How to compute fast a function and all its derivatives

Jean-Baptiste Keck

M2 MSIAM

Paper Review
02/02/2015

Introduction

- In 1983, Walter Baur and Volker Strassen expose a lower bound for derivative computation **[?]**.
- Proof is given but quite complex.
- No constructive algorithm was exposed.
- Show that we can deduce interesting bounds.

Theorem : Bauer-Strassen

The complexity of the evaluation of a rational function of several variables and all its derivatives is bounded above by three times the complexity of the evaluation of the function itself.

Contribution

Two years later in 1985, Jacques Morgenstern publish this paper **[?]** *How to compute fast a function and all its derivatives*.

- Shows an alternative proof which is much simpler.
- And in a constructive manner.
- Allows autodifferenciation techniques.

Notations

- $\mathbb{K}$ is an infinite field.
- $F \in \mathbb{K}_n(X)$ is a rational function of n variables $x_1, x_2, ..., x_n$.
- $\widetilde{F} \in \mathbb{K}_{n+1}(X)$ is a rational function of $n+1$ variables $x_1, x_2, ..., x_n, y$.
- The set of partial derivatives of F is denoted

$$F' = \left\{ \frac{\partial F}{\partial x_1}, \frac{\partial F}{\partial x_2}, ..., \frac{\partial F}{\partial x_n} \right\}.$$

Definition : Essential operation

Let $\odot$ be an arithmetical operation and $x = (x_1, x_2, ..., x_n) \in \mathbb{K}^n$.

(a $\odot$ b is an **essential operation**)

$$\Leftrightarrow \begin{cases} a = f(x) \text{ and } b = g(x) \text{ with } f, g \in \mathbb{K}_n(X) \\ \textbf{or } a \in \mathbb{K}, b = g(x) \text{ and } \odot \text{ is a division.} \end{cases}$$

Algorithm

- Let $\mathcal{A}$ be an algorithm computing F from $x = (x_1, x_2, ..., x_n)$ and $\mathbb{K}$.

Theorem/Definition

$$\exists\ u \in \mathbb{N}\ \ \mathcal{A} = \{g_1, g_2, ..., g_u\}\ \text{ with :}$$
$$\begin{cases} g_k \in \{x_1, x_2, ..., x_n\} \cup \mathbb{K} \\ \textbf{or}\ \ g_k = g_{k_1}\ \odot\ g_{k_2}\ \text{with}\ k_1, k_2 < k,\ \odot \in \{+, \times, \div\} \end{cases}$$

Example : $f(x) = f(x_1, x_2) = \frac{1+x_1^2}{x_2^4}, u = 8$

$$\left[\begin{array}{lcl} g_1 & = & 1 \\ g_2 & = & x_1 \\ g_3 & = & g_2 \times g_2 \\ g_4 & = & g_1 + g_3 \end{array}\right. \qquad \left[\begin{array}{lcl} g_5 & = & x_2 \\ g_6 & = & g_5 \times g_5 \\ g_7 & = & g_6 \times g_6 \\ g_8 & = & g_4 \div g_7 \end{array}\right.$$

Notations

Finally let :

- $s(F)$ be the number of **essential multiplications and divisions** in $\mathcal{A}$.
- $m(F)$ be the total number of **multiplications and divisions** in $\mathcal{A}$.
- $T(F)$ the total number of **essential operations** in $\mathcal{A}$.
- $\Theta(F)$ the total number of **operations** in $\mathcal{A}$.

Baur-Strassen's Theorem

With all those notations :

Baur-Strassen's Theorem

From each algorithm $\mathcal{A}$ computing F one can derive and algorithm $\mathring{\mathcal{A}}$ computing F and F' such that :

$$(P) \Leftrightarrow \begin{cases} s(F,F') & \leq & \mathbf{3}\, s(F) \\ m(F,F') & \leq & \mathbf{3}\, m(F) \\ T(F,F') & \leq & \mathbf{5}\, T(F) \\ \Theta(F,F') & \leq & \mathbf{5}\, \Theta(F) \end{cases}$$

Those inequalities are independent of the number of variables n.

Proof overview

- The proof is made by **induction on the length of the algorithm**.

Let $\mathcal{A}_u$ be an algorithm of length u computing a rational function F and g_k be the result of the first operation of $\mathcal{A}_u$.

Define $\widetilde{F}$ a function of $n+1$ variables such that $F(x_1, x_2, ..., x_n) = \widetilde{F}(x_1, x_2, ..., x_n, \mathbf{y})$ with $y = g_k(x_1, x_2, ..., x_n)$.

$\mathcal{A}_u$ induces an algorithm $\mathcal{A}_{u-1}$ which computes $\widetilde{F}$ from $x_1, x_2, ..., x_n, g_k$ and $\mathbb{K}$ in one less operation ($\mathcal{A}_{u-1}$ is of length $u-1$). **By induction hypothesis, $\widetilde{F}$ satisfies (P).**

Example of induced algorithm

Example : $F(x) = \frac{1+x_1^2}{x_2^4} \quad \Rightarrow \quad \widetilde{F}(x, g_3) = \frac{1+g_3}{x_2^4}$

$$\mathcal{A}_u \quad \Leftrightarrow \quad \left[\begin{array}{lcl} g_1 & = & 1 \\ g_2 & = & x_1 \\ g_3 & = & g_2 \times g_2 \\ g_4 & = & g_1 + g_3 \end{array}\right. \qquad \left[\begin{array}{lcl} g_5 & = & x_2 \\ g_6 & = & g_5 \times g_5 \\ g_7 & = & g_6 \times g_6 \\ g_8 & = & g_4 \div g_7 \end{array}\right.$$

$\Downarrow$

$$\mathcal{A}_{u-1} \quad \Leftrightarrow \quad \left[\begin{array}{lcl} \widetilde{g_1} & = & 1 \\ \widetilde{g_2} & = & x_1 \\ \widetilde{g_3} & = & g_3 \\ \widetilde{g_4} & = & \widetilde{g_1} + \widetilde{g_3} \end{array}\right. \qquad \left[\begin{array}{lcl} \widetilde{g_5} & = & x_2 \\ \widetilde{g_6} & = & \widetilde{g_5} \times \widetilde{g_5} \\ \widetilde{g_7} & = & \widetilde{g_6} \times \widetilde{g_6} \\ \widetilde{g_8} & = & \widetilde{g_4} \div \widetilde{g_7} \end{array}\right.$$

Proof overview

We have $$\forall h \in [\![1, n]\!] \quad \frac{\partial F}{\partial x_h} = \frac{\partial \widetilde{F}}{\partial x_h} + \frac{\partial \widetilde{F}}{\partial y} \cdot \frac{\partial g_k}{\partial x_h} \quad (1).$$

The idea is then to examine all six possible cases :

1. Case $\mathbf{g_k} = \mathbf{c} \times \mathbf{x_i}$ where $c \in \mathbb{K}$ and $i \in [\![1, n]\!]$.
2. Case $\mathbf{g_k} = \mathbf{x_i} \times \mathbf{x_j}$ where $i, j \in [\![1, n]\!]$.
3. Case $\mathbf{g_k} = \mathbf{c} \div \mathbf{x_i}$ where $c \in \mathbb{K}$ and $i \in [\![1, n]\!]$.
4. Case $\mathbf{g_k} = \mathbf{x_i} \div \mathbf{x_j}$ where $i, j \in [\![1, n]\!]$.
5. Case $\mathbf{g_k} = \mathbf{x_i} + \mathbf{c}$ where $c \in \mathbb{K}$ and $i \in [\![1, n]\!]$.
6. Case $\mathbf{g_k} = \mathbf{x_i} + \mathbf{x_j}$ where $i, j \in [\![1, n]\!]$.

Example case $g_k = c \times x_i$

First case : $g_k = c \times x_i$ where $c \in \mathbb{K}$ and $i \in [\![1, n]\!]$.

$$(1) \Rightarrow \begin{cases} \dfrac{\partial F}{\partial x_h}(x_1, ..., x_n) = \dfrac{\partial \widetilde{F}}{\partial x_h}(x_1, ..., x_n) + c \cdot \dfrac{\partial \widetilde{F}}{\partial y}(x_1, ..., x_n) & \text{if } h = i \\ \\ \dfrac{\partial F}{\partial x_h}(x_1, ..., x_n) = \dfrac{\partial \widetilde{F}}{\partial x_h}(x_1, ..., x_n) & \text{if } h \neq i \end{cases}$$

In this case we have :

$$\begin{cases} s(F, F') & = & s(\widetilde{F}, \widetilde{F}') \\ m(F, F') & \leq & m(\widetilde{F}, \widetilde{F}') + 1 + 1 \\ T(F, F') & \leq & T(\widetilde{F}, \widetilde{F}') + 1 \\ \Theta(F, F') & \leq & \Theta(\widetilde{F}, \widetilde{F}') + 1 + 1 + 1 \end{cases}$$

Example case $g_k = x_i \times x_j$

Second case : $g_k = x_i \times x_j$ where $i, j \in [\![1, n]\!]$:

$$(1) \Rightarrow \begin{cases} \dfrac{\partial F}{\partial x_h}(x_1, ..., x_n) = \dfrac{\partial \widetilde{F}}{\partial x_h}(x_1, ..., x_n) + x_j \cdot \dfrac{\partial \widetilde{F}}{\partial y}(x_1, ..., x_n) & \text{if } h = i \\ \dfrac{\partial F}{\partial x_h}(x_1, ..., x_n) = \dfrac{\partial \widetilde{F}}{\partial x_h}(x_1, ..., x_n) + x_i \cdot \dfrac{\partial \widetilde{F}}{\partial y}(x_1, ..., x_n) & \text{if } h = j \\ \dfrac{\partial F}{\partial x_h}(x_1, ..., x_n) = \dfrac{\partial \widetilde{F}}{\partial x_h}(x_1, ..., x_n) & \text{if } h \notin \{i, j\} \end{cases}$$

In this case we have :

$$\begin{cases} s(F, F') & \leq & s(\widetilde{F}, \widetilde{F}') & + & 2 & + & 1 \\ m(F, F') & \leq & m(\widetilde{F}, \widetilde{F}') & + & 2 & + & 1 \\ T(F, F') & \leq & T(\widetilde{F}, \widetilde{F}') & + & 2 & + & 2 & + & 1 \\ \Theta(F, F') & \leq & \Theta(\widetilde{F}, \widetilde{F}') & + & 2 & + & 2 & + & 1 \end{cases}$$

Putting everything together

We do exactly the same for the 4 other cases :

$g_k = c \dotdiv x_i \quad g_k = x_i \dotdiv x_j \quad g_k = x_i + c \quad g_k = x_i + x_j$

Putting everything together and using (P) for $\widetilde{F}$, we get at most the inequalities:

$$\begin{cases} s(F,F') & \leq & s(\widetilde{F},\widetilde{F}') & + & 3 & \leq & 3\,s(\widetilde{F}) & + & 3 & = & 3\,s(F) \\ m(F,F') & \leq & m(\widetilde{F},\widetilde{F}') & + & 3 & \leq & 3\,m(\widetilde{F}) & + & 3 & = & 3\,m(F) \\ T(F,F') & \leq & T(\widetilde{F},\widetilde{F}') & + & 5 & \leq & 5\,T(\widetilde{F}) & + & 5 & = & 5\,T(F) \\ \Theta(F,F') & \leq & \Theta(\widetilde{F},\widetilde{F}') & + & 5 & \leq & 5\,\Theta(\widetilde{F}) & + & 5 & = & 5\,\Theta(F) \end{cases}$$

$\Rightarrow$ (P) for F, which proves the theorem since the beginning of the induction is trivial.

Those inequalities are independent of the number of variables n since we just added a bounded number of new operations.

Conclusion

- New constructive proof for the Baur-Strassen Theorem.
- Exhibits a constructive recursive backward way to generate partial derivatives algorithm.
- Establish the basis of automatic differentiation (algorithmic differentiation).

Extension :

- Add elementary functions ($exp, log, sin, cos, \ldots$) to elementary arithmetic operations ($+, -, \times, \div$).
- Every computer program, no matter how complicated, executes a sequence of elementary arithmetic operations.

$\Rightarrow$ Derivatives of arbitrary order can be computed automatically, using at most a small constant factor more arithmetic operations than the original program.

References

- Jacques Morgenstern.
  How to compute fast a function and all its derivatives.
  ACM SIGACT News, 16:60–62, 1985.
- Walter Baur; Volker Strassen.
  The complexity of partial derivatives.
  Theoretical Computer Science, 22:317–330, 1983.

Any questions ?